

Fig. 1

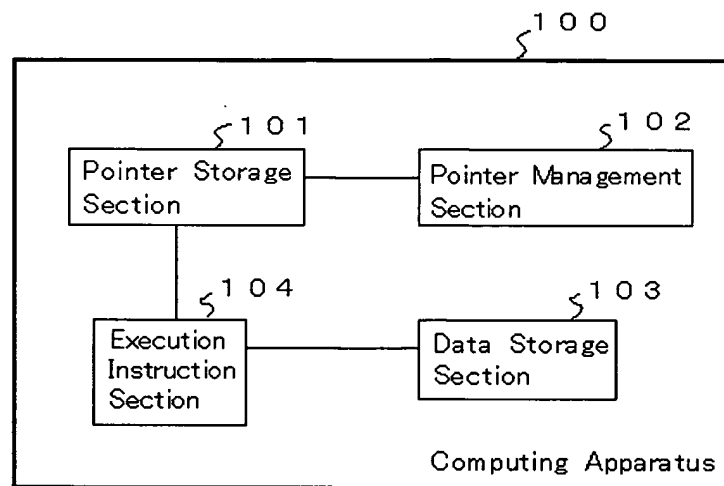


Fig. 2

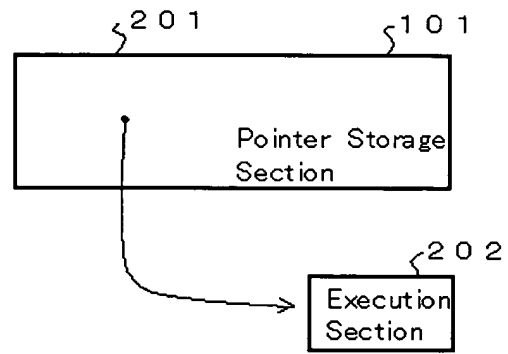


Fig. 3

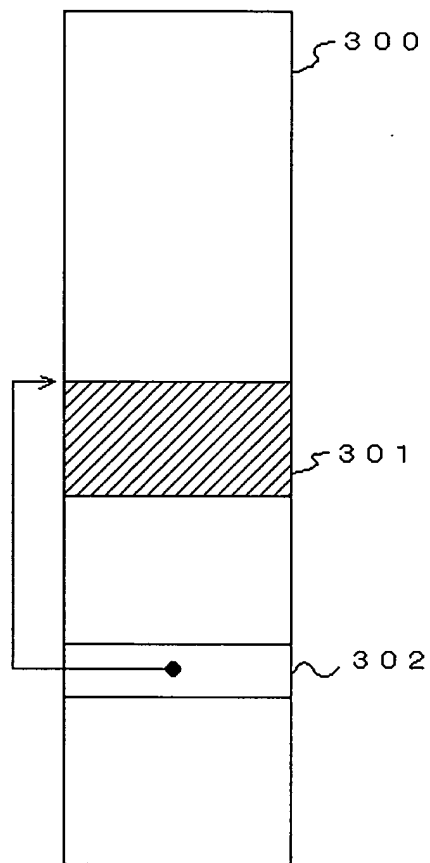


Fig 4

```
void (*fp)();          . . . (a)
```

```
void                . . . (b)
f(dp)
void *dp;
{
    :
}
```

```
main()                . . . (c)
{
    :
    fp = f;           . . . (d)
    :
}
```

Fig. 5

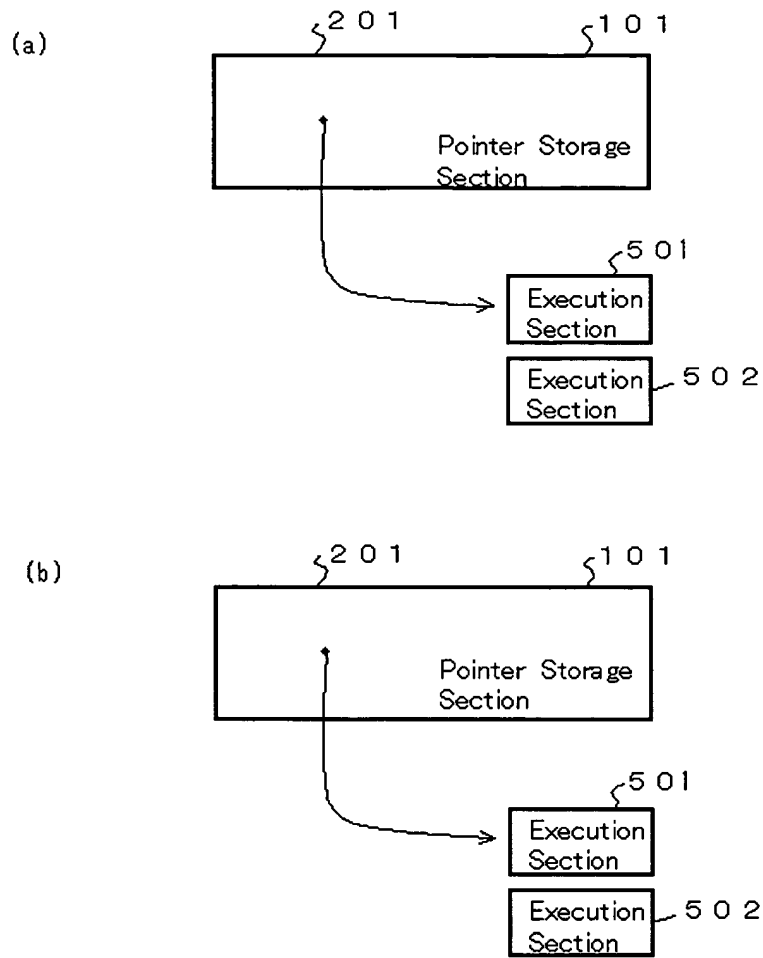


Fig 6

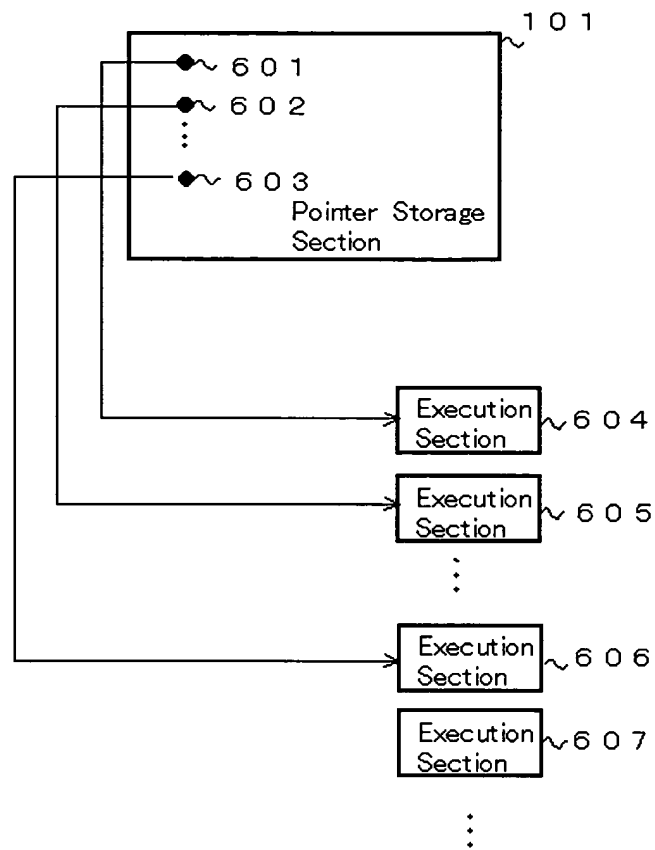


Fig. 7

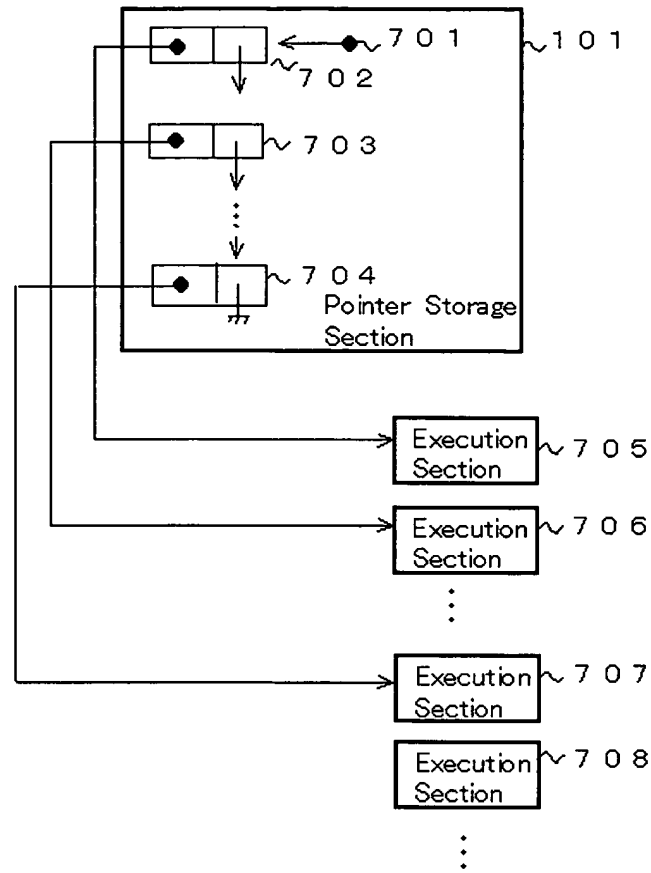


Fig. 8

```
struct pointerlist {  
    void (*fp)();          . . .      (a)  
    struct pointerlist *next; . . .    (b)  
} *pointerlistbase;
```


Fig 9

```
void *p ;                . . .      (a)
```

◆
◆
◆

P = (void*) 0x37468AB8; . . . (b)

◆◆◆

$$(*fp)(p); \quad \cdot \cdot \cdot \quad (c)$$

Fig. 10

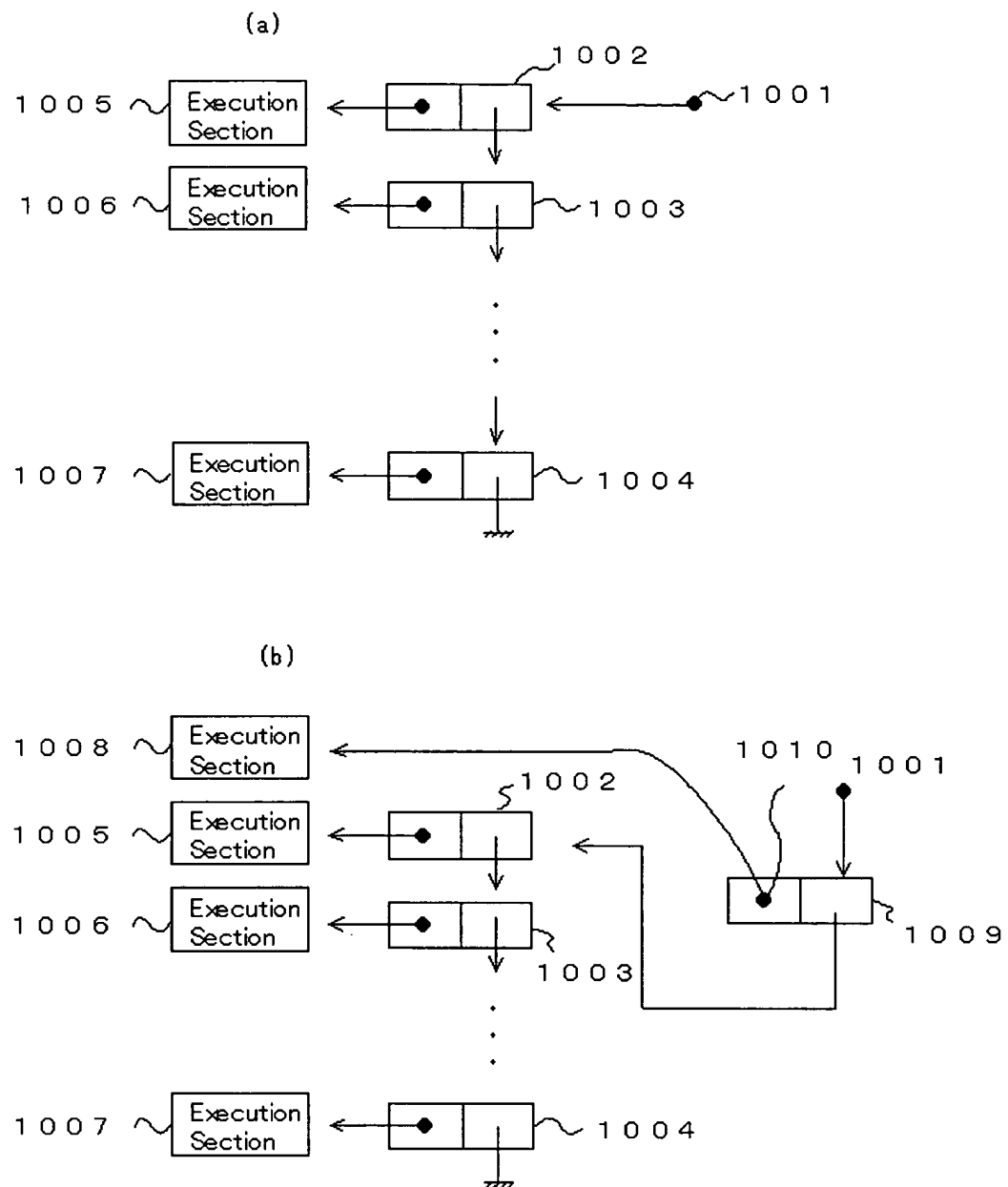


Fig. 1 1

```
struct pointerlist *pl;          . . . (a)
```

```
pl = malloc (sizeof *pl);        . . . (b)
```

```
pl -> fp = g;
pl -> next = pointerlistbase;
pointerlistbase = pl ;           } (c)
```

Fig. 1 2

```
struct pointerlist *pl;          . . .      (a)
```

```
pl = pointerlistbase;  
pointerlistbase = pointerlistbase -> next; } (b)
```

```
free (pl);                      . . .      (c)
```

Fig 1 3

```
struct pointerlist *pl;
int                n;
void               *p;
    }              . . . (a)

P = (void*) 0x37468AB8;      . . . (b)
pl = pointerlistbase;       . . . (c)
for ( n = 0; n < 3; n++)    }
    pl = pl -> next;        . . (d)

( *pl -> fp)(p);             . . . (e)
```

Fig 1 4

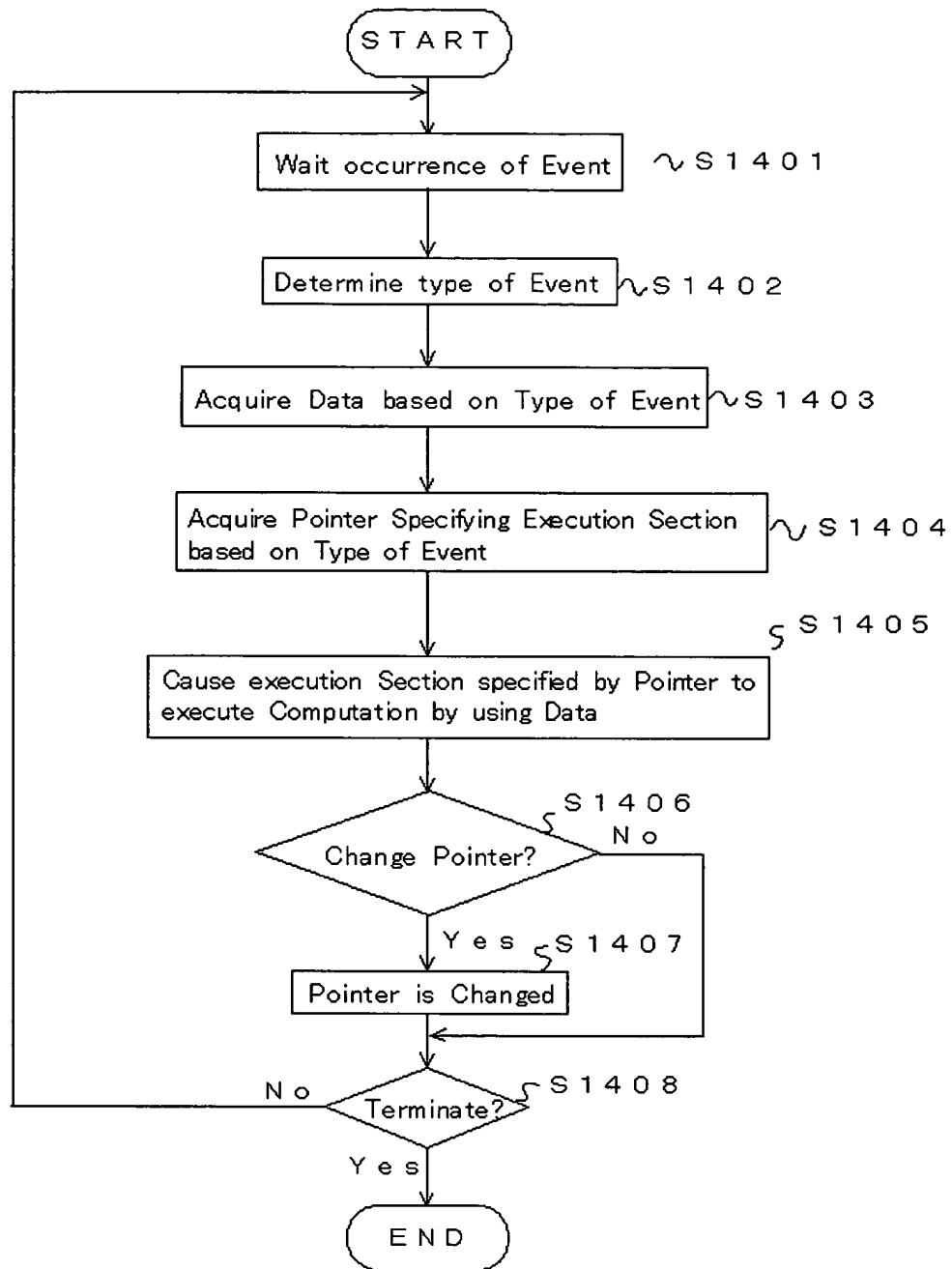
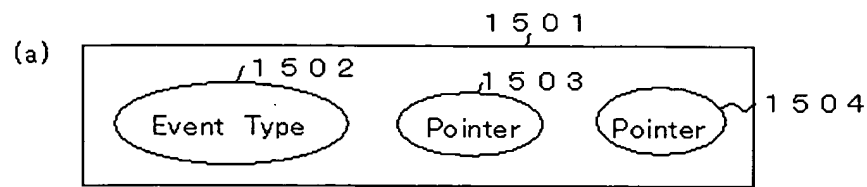


Fig 1 5



(b)

```
struct {  
    enum eventType et;  
    void (*fp)();  
    void *dp;  
};
```

Fig. 1 6

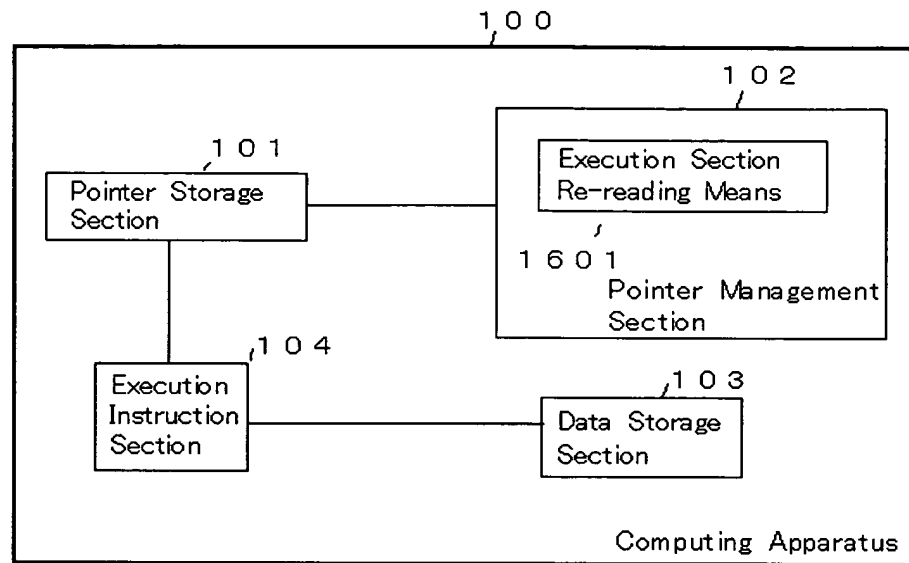


Fig. 1 7

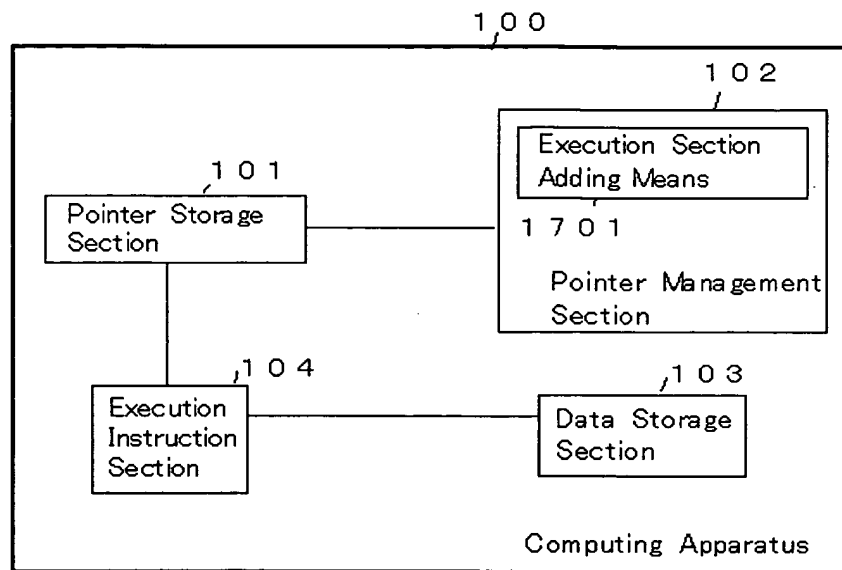


Fig. 1 8

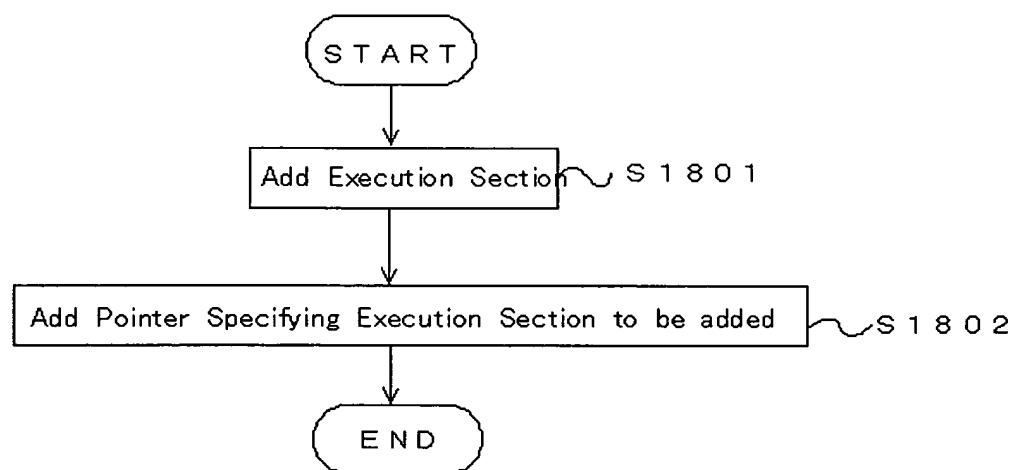


Fig. 1 9

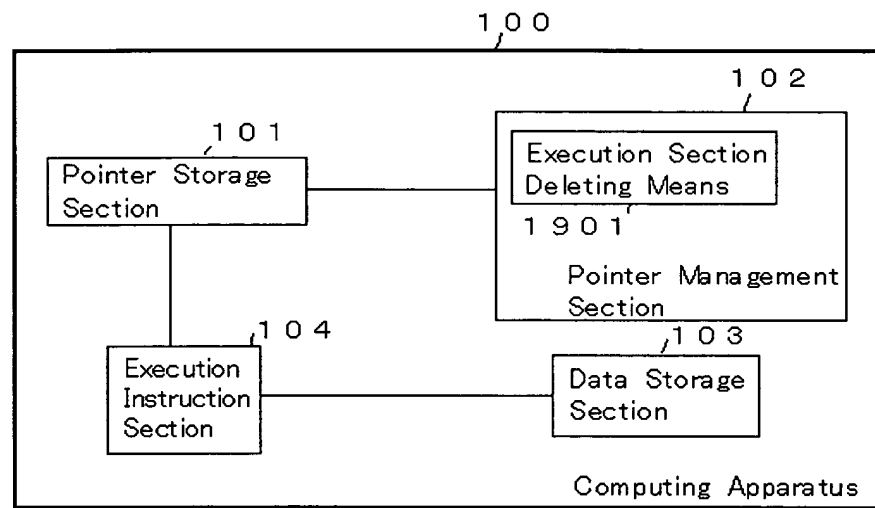


Fig 2 0

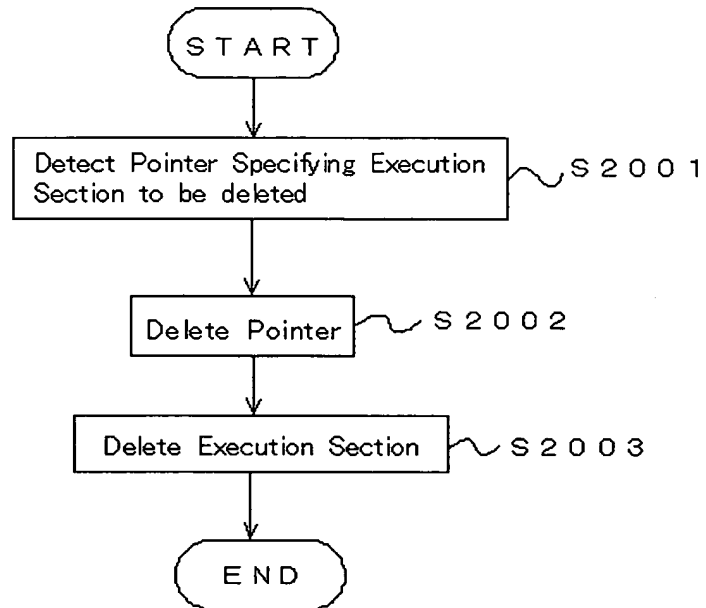


Fig 2 1

```
struct pointerlist *pl;    }    . . . (a)
void (*ep)();
```



```
ep = (void (*)()) 0x284729EC;    . . . (b)
pl = pointerlistbase;    . . . (c)
while ( pl != NULL ) {
    if ( pl -> fp == ep )    }    . . . (d)
        break;
    pl = pl -> next;    . . . (e)
}
```

Fig 2 2

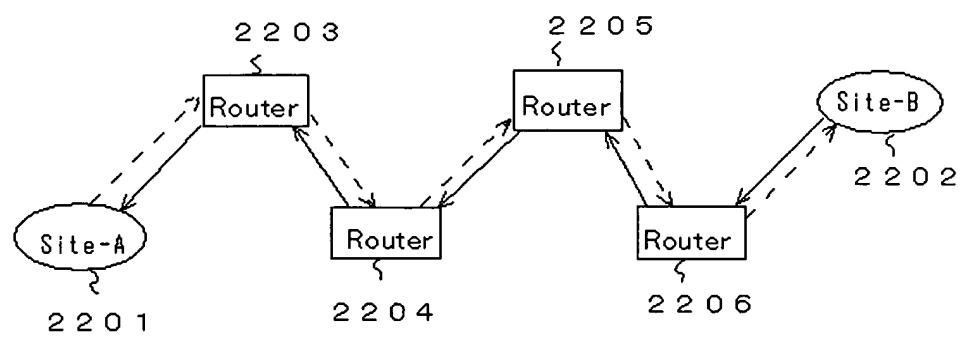


Fig 2 3

